

This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1. (Currently amended) A method for implementing a bimodal virtual device in a computer system, said method comprising:
 - ~~utilizing~~ configuring the bimodal virtual device such that it selectively operates with one or more virtual machines in two different modes,
 - a first mode comprising ~~[[as-]]~~ a hardware mode during which the bimodal virtual device emulates a specific hardware device and is ~~accessible~~ accessed by a virtual machine via a device driver that is ~~capable~~ configured to of driving drive the specific hardware device, and
 - a second mode comprising an idealized mode where the bimodal virtual device ~~operates with improved performance over the first mode~~ is optimized for a virtualized environment and wherein said virtual device operates without emulating the specific hardware device.
2. (Previously presented) The method of claim 1 wherein:
 - the bimodal virtual device selectively operates in the hardware mode when a device driver interfacing with said bimodal virtual device has not been designed to interface with said bimodal virtual device operating in said second mode; and
 - the bimodal virtual device selectively operates in the idealized mode when the driver interfacing with said bimodal virtual device has been designed to interface with said bimodal virtual device operating in said second mode.
3. (original) The method of claim 2 wherein the functionality of the second mode extends the functionality of the first mode.
4. (original) The method of claim 2 wherein the functionality of the second mode independent of the functionality of the first mode.

5. (original) The method of claim 4 wherein the functionality of the second mode functionality of the first mode.

6. (original) The method of claim 4 wherein the functionality of the second mode portions of the functionality of the first mode.

7. (original) The method of claim 2 wherein the second mode is enabled through least one bit in a virtual device register.

8. (original) The method of claim 2 wherein the second mode is enabled through least one bit in a register specifically created for utilization by one or more virtual

9. (original) The method of claim 2 wherein the second mode is enabled through prescribed sequence of commands or data that change a value in at least one register

10. (original) The method of claim 2 wherein
the second mode is enabled through the use of a second mode driver instilled within a guest operating system environment; and
if the second mode driver is not present, a first mode driver is instead enabled

11. (Currently amended) A computer system, said computer system comprising a bimodal virtual device that selectively operates as:
a hardware virtual device in a first mode; and
as an idealized virtual device in a second mode[[],] ;
wherein in the first mode the bimodal virtual device emulates a real hardware device, and in the second mode the bimodal virtual device functions as an abstract device that is a same type of device as the real hardware device but is incompatible with software configured to interact with the real hardware device.

12. (Previously presented) The system of claim 11 wherein:
the bimodal virtual device selectively operates when a driver interfacing with said bimodal virtual device has not been designated to interface with said abstract device; and

the bimodal virtual device selectively operates in the second mode when the driver interfacing with said bimodal virtual device has been designated to interface with said bimodal virtual device operating in said second mode.

13. (original) The system of claim 12 wherein the functionality of the second mode extends the functionality of the first mode.

14. (original) The system of claim 12 wherein the functionality of the second mode is independent of the functionality of the first mode.

15. (original) The system of claim 14 wherein the functionality of the second mode disables the functionality of the first mode.

16. (original) The system of claim 14 wherein the functionality of the second mode disables portions of the functionality of the first mode.

17. (original) The system of claim 12 wherein the second mode is enabled through the use of at least one bit in a virtual device register.

18. (original) The system of claim 12 wherein the second mode is enabled through the use of at least one bit in a register specifically created for utilization by one or more virtual devices.

19. (original) The system of claim 12 wherein the second mode is enabled through the use of a prescribed sequence of commands or data that change a value in at least one register.

20. (original) The system of claim 12 wherein the second mode is enabled through the use of a second mode driver installed within a guest operating system environment; and

21. (Currently amended) A computer system, said computer system comprising: a virtual machine environment; and

a bimodal virtual device that selectively operates as a hardware virtual device in a first mode as an idealized virtual device in a second mode with said virtual machine environment, where in the first mode the bimodal virtual device emulates a real hardware device, and in the second mode the bimodal virtual device functions as an abstract device that is a same type of device as the real hardware device but is incompatible with software configured to interact with the real hardware device.

22. (Previously presented) The system of claim 21 wherein:
the bimodal virtual device selectively operates in the first mode when a driver interfacing with said bimodal virtual device has not been designated to interface with said bimodal device operating in said second mode; and
the bimodal virtual device selectively operates in the second mode when the driver interfacing with said bimodal virtual device has been designated to interface with said bimodal virtual device operating in said second mode.

23. (original) The system of claim 22 wherein the functionality of the second mode extends the functionality of the first mode.

24. (original) The system of claim 22 wherein the functionality of the second mode is independent of the functionality of the first mode.

25. (original) The system of claim 24 wherein the functionality of the second mode disables the functionality of the first mode.

26. (original) The system of claim 24 wherein the functionality of the second mode disables portions of the functionality of the first mode.

27. (original) The system of claim 22 wherein the second mode is enabled through the use of at least one bit in a virtual device register.

28. (original) The system of claim 22 wherein the second mode is enabled through the use of at least one bit in a register specifically created for utilization by one or more

virtual devices.

29. (original) The system of claim 22 wherein the second mode is enabled through the use of a prescribed sequence of commands or data that changes a value in at least one register.

30. (original) The system of claim 22 wherein
the second mode is enabled through the use of a second mode driver installed within a guest operating system environment; and
if the second mode driver is not present, a first mode driver is instead enabled.

31. (Previously presented) A computer-readable medium comprising computer-readable instructions, said computer-readable instructions comprising instructions for a bimodal virtual device to selectively operate as a hardware virtual device in a first mode and as an idealized virtual device in a second mode, where in the first mode the bimodal virtual device emulates a real hardware device, and in the second mode the bimodal virtual device functions as an abstract device that is a same type of device as the real hardware device but is incompatible with software configured to interact with the real hardware device.

32. (Previously presented) The computer-readable instructions of claim 21 further comprising instructions for:

the bimodal virtual device to selectively operate in the first mode when a driver interfacing with said bimodal virtual device has not been designated to interface with said bimodal virtual device operating in said second mode; and

the bimodal virtual device to selectively operate in the second mode when the driver interfacing with said bimodal virtual device has been designated to interface with said bimodal virtual device operating in said second mode.

33. (Original) The computer-readable instructions of claim 32 further comprising instructions for the functionality of the second mode to extend the functionality of the first mode.

34. (Original) The computer-readable instructions of claim 32 further comprising instructions for the functionality of the second mode that are separate and distinct from instructions for the functionality of the first mode.

35. (Original) The computer-readable instructions of claim 34 further comprising instructions for the second mode to disable the functionality of the first mode.

36. (Original) The computer-readable instructions of claim 34 further comprising instructions for the second mode to disable portions of the functionality of the first mode.

37. (Original) The computer-readable instructions of claim 32 further comprising instructions for enabling the second mode through the use of at least one bit in a virtual device register.

38. (Original) The computer-readable instructions of claim 32 further comprising instructions for enabling the second mode through the use of at least one bit in a register specifically created for utilization by one or more virtual devices.

39. (Original) The computer-readable instructions of claim 32 further comprising instructions for enabling the second mode through the use of a prescribed sequence of commands or data that change a value in at least one register.

40. (Original) The computer-readable instructions of claim 32 further comprising instructions for:

enabling the second mode through the use of a second mode driver installed within a guest operating system environment; and

if the second mode driver is not present, enabling a first mode through the use of a first mode driver.